

Sampark Setu API Documentation

This document provides comprehensive documentation for all API endpoints used in the Sampark Setu project.

Base Configuration

- **Base URL:** `http://localhost:3000/api`
- **Content-Type:** `application/json`
- **Authentication:** Bearer Token (JWT)

Authentication

All API requests require authentication via Bearer token in the Authorization header:

```
Authorization: Bearer <your_jwt_token>
```

1. Authentication APIs

1.1 Login

Endpoint: `POST /login`

Description: Authenticate user and get access token

Request Body:

```
{
  "user_name": "string",
  "password": "string"
}
```

Response Body:

```
{
  "success": true,
  "message": "Login successful",
  "statusCode": 200,
  "data": {
    "token": "string",
    "role": "string",
    "region_id": 123
  }
}
```

1.2 Get Current User

Endpoint: `GET /users/me`

Description: Get current authenticated user details

Request Body: None

Response Body:

```
{
  "success": true,
  "message": "User details retrieved",
  "statusCode": 200,
  "data": {
    "user_id": 123,
    "user_name": "string",
    "role": "string",
    "region_id": 123,
    "is_active": 1,
    "created_at": "2024-01-01T00:00:00Z",
    "updated_at": "2024-01-01T00:00:00Z",
    "region_details": {
      "prant": {
        "region_id": 1,
        "name": "string",
        "code": "string",
        "type": "PRANT"
      },
      "vibhag": {
        "region_id": 2,
        "name": "string",
        "code": "string",
        "type": "VIBHAG"
      },
      "jila": {
        "region_id": 3,
        "name": "string",
        "code": "string",
        "type": "JILA"
      },
      "nagar": {
        "region_id": 4,
        "name": "string",
        "code": "string",
        "type": "NAGAR"
      }
    }
  }
}
```

2. Region Management APIs

2.1 Get Regions

Endpoint: GET /v1/regions

Description: Get regions by type and optional parent ID

Query Parameters:

- `type` (required): Region type - `PRANT` , `VIBHAG` , `JILA` , `NAGAR` , or `KHAND`
- `parent_id` (optional): Parent region ID for hierarchical queries

Request Body: None

Response Body:

```
{
  "success": true,
  "message": "Regions retrieved",
  "statusCode": 200,
  "data": [
    {
      "region_id": 123,
      "name": "string",
      "code": "string",
      "type": "PRANT",
      "parent_id": null,
      "created_at": "2024-01-01T00:00:00Z",
      "updated_at": "2024-01-01T00:00:00Z"
    }
  ]
}
```

2.2 Get Region by ID

Endpoint: `GET /v1/regions/{regionId}`

Description: Get specific region details by ID

Path Parameters:

- `regionId` : Region ID

Request Body: None

Response Body:

```
{
  "success": true,
  "message": "Region details retrieved",
  "statusCode": 200,
  "data": {
    "region_id": 123,
    "name": "string",
    "code": "string",
    "type": "PRANT",
    "parent_id": null,
    "created_at": "2024-01-01T00:00:00Z",
    "updated_at": "2024-01-01T00:00:00Z"
  }
}
```

2.3 Fetch Regions

****Endpoint**:** `GET /fetch-regions`

****Description**:** Fetch regions with region_id and ignore_validation parameters

****Query Parameters**:**

- `region_id` (required): Region ID
- `ignore_validation` (optional): "yes" or "no" (default: "yes")

****Request Body**:** None

****Response Body**:**

````json

```
{
 "success": true,
 "data": {
 "region_name": "string",
 "region_type": "prant",
 "child_region_names": ["string"],
 "child_regions": {
 "vibhag": [
 {
 "id": 1,
 "name": "string",
 "code": "string"
 }
],
 "jila": [
 {
 "id": 2,
 "name": "string",
 "code": "string"
 }
],
 "nagar": [
 {
 "id": 3,
 "name": "string",
 "code": "string"
 }
],
 "khand": [
 {
 "id": 4,
 "name": "string",
 "code": "string"
 }
]
 }
 }
}
```

```
}
}
```

## 3. Toli Management APIs

### 3.1 Create Toli

**Endpoint:** POST /toli

**Description:** Create a new toli (group)

**Request Body:**

```
{
 "name": "string",
 "type": "PRANT",
 "region_id": 123,
 "toli_user_id": 456,
 "pramukh": {
 "name": "string",
 "mobile": "string"
 },
 "members": [
 {
 "name": "string",
 "mobile": "string"
 }
]
}
```

**Response Body:**

```
{
 "success": true,
 "message": "Toli created successfully",
 "statusCode": 200,
 "data": {
 "toli_id": 123,
 "status": "created"
 }
}
```

### 3.2 Get Tolis

**Endpoint:** GET /toli

**Description:** Get list of tolis with optional filtering

**Query Parameters:**

- region\_id (optional): Filter by region ID

- `type` (optional): Filter by toli type
- `limit` (optional): Number of records to return
- `offset` (optional): Number of records to skip

**Request Body:** None

**Response Body:**

```
{
 "success": true,
 "message": "Tolis retrieved",
 "statusCode": 200,
 "data": [
 {
 "tolli_id": 123,
 "name": "string",
 "type": "PRANT",
 "region_id": 123,
 "tolli_user_id": 456,
 "pramukh_json": {
 "name": "string",
 "mobile": "string"
 },
 "members_json": [
 {
 "name": "string",
 "mobile": "string"
 }
],
 "created_at": "2024-01-01T00:00:00Z",
 "updated_at": "2024-01-01T00:00:00Z"
 }
]
}
```

---

## 4. Person Management APIs (Utsuk Shakti)

### 4.1 Create Person

**Endpoint:** POST `/person`

**Description:** Create a new person record

**Request Body:**

```
{
 "name": "string",
 "phone_number": "string",
 "email": "string",
 "sex": "MALE",
 "address_text": "string",
 "region_id": 123,
```

```
"visheshta": "string",
"answers": {
 "key1": "value1",
 "key2": "value2"
}
}
```

**Response Body:**

```
{
 "success": true,
 "message": "Person created successfully",
 "statusCode": 200,
 "data": {
 "person_id": 123,
 "status": "created"
 }
}
```

## 4.2 Get Persons

**Endpoint:** GET /person

**Description:** Get list of persons with optional filtering

**Query Parameters:**

- `region_id` (optional): Filter by region ID
- `search` (optional): Search term for name/phone/email
- `limit` (optional): Number of records to return
- `offset` (optional): Number of records to skip

**Request Body:** None

**Response Body:**

```
{
 "success": true,
 "message": "Persons retrieved",
 "statusCode": 200,
 "data": [
 {
 "person_id": 123,
 "name": "string",
 "phone_number": "string",
 "email": "string",
 "sex": "MALE",
 "address_text": "string",
 "region_id": 123,
 "created_by": 456,
 "visheshta": "string",
 "answers_json": {
```

```
 "key1": "value1",
 "key2": "value2"
 },
 "created_at": "2024-01-01T00:00:00Z",
 "updated_at": "2024-01-01T00:00:00Z"
}
]
```

## 5. Visit Management APIs (Parivar Data)

### 5.1 Create Visit

**Endpoint:** POST /visit

**Description:** Create a new visit record

**Request Body:**

```
{
 "person_name": "string",
 "person_phone": "string",
 "person_email": "string",
 "person_sex": "MALE",
 "toli_id": 123,
 "region_id": 456,
 "visited_at": "2024-01-01T00:00:00Z",
 "total_members": 5,
 "male_count": 3,
 "female_count": 2,
 "kids_count": 1,
 "nishulk_sticker": 10,
 "nishulk_folder": 5,
 "nishulk_books": 3,
 "shashulk_pushtak": 2,
 "address_text": "string",
 "notes": "string"
}
```

**Response Body:**

```
{
 "success": true,
 "message": "Visit created successfully",
 "statusCode": 200,
 "data": {
 "visit_id": 123,
 "status": "created"
 }
}
```



## 5.2 Get Visits

**Endpoint:** GET /visit

**Description:** Get list of visits with optional filtering

**Query Parameters:**

- `region_id` (optional): Filter by region ID
- `tol_i_id` (optional): Filter by toli ID
- `user_id` (optional): Filter by user ID
- `start_date` (optional): Filter visits from this date
- `end_date` (optional): Filter visits until this date
- `search` (optional): Search term for person name/phone/email
- `limit` (optional): Number of records to return
- `offset` (optional): Number of records to skip

**Request Body:** None

**Response Body:**

```
{
 "success": true,
 "message": "Visits retrieved",
 "statusCode": 200,
 "data": [
 {
 "visit_id": 123,
 "person_name": "string",
 "person_phone": "string",
 "person_email": "string",
 "person_sex": "MALE",
 "tol_i_id": 123,
 "region_id": 456,
 "visited_at": "2024-01-01T00:00:00Z",
 "created_by": 789,
 "total_members": 5,
 "male_count": 3,
 "female_count": 2,
 "kids_count": 1,
 "nishulk_sticker": 10,
 "nishulk_folder": 5,
 "nishulk_books": 3,
 "shashulk_pushtak": 2,
 "address_text": "string",
 "notes": "string",
 "created_at": "2024-01-01T00:00:00Z"
 }
]
}
```

---

## Error Responses

All APIs return standardized error responses:

```
{
 "success": false,
 "message": "Error description",
 "statusCode": 400,
 "data": null
}
```

### Common HTTP Status Codes:

- 200 : Success
  - 400 : Bad Request
  - 401 : Unauthorized (Token expired/invalid)
  - 403 : Forbidden
  - 404 : Not Found
  - 500 : Internal Server Error
- 

## Authentication Flow

1. User submits credentials to /login
  2. Server returns JWT token and user details
  3. Client stores token in localStorage
  4. All subsequent requests include token in Authorization header
  5. On 401 response, client automatically redirects to login page
- 

## Notes

- All timestamps are in ISO 8601 format
- Region types are case-sensitive: PRANT , VIBHAG , JILA , NAGAR , KHAND
- Sex values are: MALE , FEMALE , OTHER , UNSPECIFIED
- All optional fields can be null in responses
- Pagination is supported via limit and offset parameters